

**VŠB - Technická univerzita Ostrava**  
**Fakulta elektrotechniky a informatiky**  
**Katedra kybernetiky a biomedicínského inženýrství**

**Mobilní technologie s podporou konceptu internet of  
things**

**Mobile Technologies Internet of Things Concept Support**

## Zadání bakalářské práce

Student: **Hana Gregorová**  
Studijní program: **B2649 Elektrotechnika**  
Studijní obor: **3901R039 Biomedicínský technik**  
Téma: **Mobilní technologie s podporou konceptu internet of things  
Mobile Technologies Internet of Things Concept Support**

### Zásady pro vypracování:

Téma této bakalářské práce sleduje stále aktuálnější koncept Internet Of Things (IOT), který v blízké budoucnosti nalezne své uplatnění v oblasti logistiky, která dnes zasahuje skoro do všech odvětví průmyslu, zdravotnictví i zemědělství.

Předmět, věc nebo i živé zvíře či člověk mohou představovat entitu, jednotku, která je označena jednoduchým a ekonomicky nenáročným identifikátorem, který se nazývá NFC (Near Field Communication) tag ať už pasivní či aktivní. Tyto tagy vycházejí z technologie RFID (Radio Frequency Identification). Tento identifikátor v sobě nese informaci o stavu dané věci, či člověka. V případě zdravotnictví tento tag může obsahovat informaci o vyšetřeních, které daný pacient absolvoval, nebo léčích, které mu mají být dodávány apod. Další možnou aplikací v oblasti medicíny může být evidence zdravotnického materiálu, ať už operační náčiní nebo krevní konzervy. V případě logistiky zboží mohou dnes velké obchodní řetězce sledovat stav jednotlivých kusů zboží včetně informací o původu a cestě různými skladišti (čímž je zajištěna ochrana proti krádeži, nebo přetištění data výroby), v případně aktivních tagů také teplotu, v níž byly výrobky uchovány. Student má v této práci za úkol vytvořit systém na mobilním zařízení typu smartphone s operačním systémem Android, které bude schopno identifikovat a přečíst informace z označené entity. Entita může být označena čárovým či 2D maticovým (QR) kódem nebo RFID tagem.

1. Seznámení se s programovacím jazykem Java a jeho využití v prostředí OS android.
2. Seznámení se s prostředky k ovládání vestavěné videokamery.
3. Prostudování a seznámení se s možnostmi zpracování identifikačních technologií jako jsou QR kódy, čárové kódy, RFID - NFC pomocí knihovny zxing.
4. Vytvoření aplikace pro čtení a zápis NFC tagů pro mobilního telefon s OS Android.
5. Vytvoření uživatelské aplikace pro mobilní zařízení, která propojuje informace o entitách s NFC tagy a databází určenou pro uchovávání těchto informací.
6. Zhodnocení dosažených výsledků.

### Seznam doporučené odborné literatury:

- 1] MURPHY, Mark L. *Android 2 Průvodce programováním mobilních aplikací*. překlad Jakub MUŽÍK. 1.vyd. Brno:COMPUTER PRESS, 2011. ISBN 978-80-251-3194-7.
- [2] MEIER, Reto. *Professional Android Application Development*. Indianapolis(USA): Wiley Publishing Inc., 2009. ISBN 978-0-470-34471-2. Dostupné také z:  
[http://www.google.cz/url?sa=t&rct=j&q=professional%20android%20application%20development&source=web&cd=2&ved=0CEAQFjAB&url=http%3A%2F%2Fandroidteam.googlecode.com%2Ffiles%2Fprofessional\\_android\\_application\\_development.pdf&ei=G-UFT7DyD8\\_IswaXqumADw&usq=AFQjCNFyzLrx89xFrFq3rvWtSjPohcvQA&sig2=ynMy9esnk8kst9B](http://www.google.cz/url?sa=t&rct=j&q=professional%20android%20application%20development&source=web&cd=2&ved=0CEAQFjAB&url=http%3A%2F%2Fandroidteam.googlecode.com%2Ffiles%2Fprofessional_android_application_development.pdf&ei=G-UFT7DyD8_IswaXqumADw&usq=AFQjCNFyzLrx89xFrFq3rvWtSjPohcvQA&sig2=ynMy9esnk8kst9B)

5kCTcKg

[3] ECKEL, Bruce. *Myslime v jazyku Java: knihovna programátora*. Praha: Grada Publishing, 2001. ISBN 80-247-9010-6.

[4] WIKIPEDIA, the Free Encyclopedia. *NFC, Near Field Communication*. [online] [cit. 2012-10-16]. Dostupné z: [http://en.wikipedia.org/wiki/Near\\_field\\_communication](http://en.wikipedia.org/wiki/Near_field_communication).

[5] ANDROID DEVELOPERS. *The Developer's Guide* [online] [cit. 2012-10-12]. Dostupné z: <http://developer.android.com/guide/index.html>.

[6] ČESKO.GS1 CZECH REPUBLIC. *GS1 Czech Republic*. Publikace vytvořeny organizací GS1 [online], c2012. Dostupné z: <http://www.gs1cz.org/sluzby/publikace/>.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Jakub Jirka**

Datum zadání: 16.11.2012

Datum odevzdání: 07.05.2013



doc. Ing. Jiří Kozíorek, Ph.D.  
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.  
děkan fakulty

## **Prohlášení**

„Prohlašuji, že jsem tuto bakalářskou práci vypracovala samostatně. Uvedla jsem všechny literární prameny a publikace, ze kterých jsem čerpala.“



Datum odevzdání: 6. 5. 2013

Hana Gregorová

## **Poděkování**

Děkuji svému vedoucímu bakalářské práce panu Ing. Jakubu Jirkovi, za jeho ochotu a vstřícnost, který mi napomohl k vytvoření této bakalářské práce.

Také bych chtěla poděkovat své rodině, která mi byla oporou při studiu na Vysoké škole Báňské – Technické univerzitě Ostrava.

## **Abstrakt**

Cílem této bakalářské práce je vytvoření mobilní aplikace, která bude schopna načíst čárové kódy a RFID tagy. Tato označení by měla sdělovat informace o pacientovi, které bude možné ukládat do databáze i sledovat v databázi.

Tato práce byla řešená v prostředí Eclipse s pluginem ADT, která užívá jazyk Java. Dále byla aplikace testována na mobilním zařízení LG Optimus One P500, Samsung Google Nexus S i9023 a emulátoru.

## **Klíčová slova**

NFC, QR, RFID, EAN, OS, eclipse, operační systém, emulátor

## **Abstract**

In this Bachelor thesis aims to create a mobile application that will be able to read barcodes and RFID tags. These labels should disclose patient information that can be stored in the database and monitor the database.

This work has been addressed in Eclipse with ADT plugin, which uses a Java language. Furthermore, the application was tested on a mobile device, LG Optimus One P500, Samsung and Google emulator.

## **Key words**

NFC, QR, RFID, EAN, OS, eclipse, operating system, emulátor

## Seznam použitých zkratek

NFC	Near field communication
OS	Operační systém
PDA	Personal digital assistant
SMS	Short message service
DVM	Dalvik Virtual Machine
API	Application Programming Interface
LIFO	Last in, first out
EAN	European Article Number
QR	Quick Response Code
RFID	Radio Frequency Identification

# Obsah

<b>1. Úvod .....</b>	<b>- 1 -</b>
<b>2. Operační systém Android.....</b>	<b>- 2 -</b>
2.1 Vývoj OS.....	- 2 -
2.2 Architektura OS.....	- 3 -
2.3 Android aplikace.....	- 4 -
2.4 Software Development Kit (SDK).....	- 6 -
<b>3. Identifikační technologie .....</b>	<b>- 7 -</b>
3.1 Čárové kódy .....	- 7 -
3.1.1 Jednodimenzionální kódy.....	- 7 -
3.1.1.1 EAN kód .....	- 8 -
3.1.2 Dvoudimenzionální kódy .....	- 9 -
3.1.2.1 QR kód.....	- 9 -
3.1.3 Třidimenzionální kódy .....	- 10 -
3.2 Knihovna Zxing .....	- 11 -
3.3 RFID .....	- 11 -
3.3.1 Komponenty RFID systému .....	- 12 -
3.3.2 Rozdělení RFID systému podle frekvenčního pásma .....	- 12 -
3.3.3 Rozdělení RFID tagů.....	- 13 -
<b>4. Analýza, návrh a implementace aplikace .....</b>	<b>- 14 -</b>
4.1 Základní požadavky .....	- 14 -
4.2 Datová analýza .....	- 14 -
4.3 Funkční analýza.....	- 15 -
4.4 Návrh uživatelského prostředí .....	- 17 -
4.5 Návrh řešení .....	- 18 -
4.6 Implementace .....	- 19 -
<b>5. Uživatelská dokumentace .....</b>	<b>- 20 -</b>
5.1. Aktivní aplikace.....	- 20 -
5.2. Scan a NFC systém .....	- 21 -
5.3. Databáze.....	- 22 -
5.4. Upozornění.....	- 23 -



<b>6.</b>	<b>Programátorská dokumentace .....</b>	<b>- 24 -</b>
6.1	Struktura aplikace .....	- 24 -
6.1.1	Struktura adresáře .....	- 25 -
6.2	Struktura tříd.....	- 27 -
6.2.1	Hlavní aktivity, třídy .....	- 27 -
6.2.2	Použité metody a funkce .....	- 27 -
6.2.3	Výjimky .....	- 28 -
<b>7.</b>	<b>Závěr.....</b>	<b>- 29 -</b>
	<b>Literatura .....</b>	<b>- 30 -</b>
	<b>Seznam příloh .....</b>	<b>- 32 -</b>
	<b>Obsah CD .....</b>	<b>- 32 -</b>

# 1. Úvod

V dnešní době se čím dál více využívá konceptu Internet Of Things (IOT) a to nejen v oblasti technického průmyslu nebo zemědělství, ale také ve zdravotnictví. Předměty, věci a nebo zvířata či lidé mohou představovat entitu, jednotku, která se může označit jednoduchým a ekonomicky nenáročným identifikátorem, který se nazývá NFC (Near Field Communication) tag. Tento identifikátor v sobě nese určitou informaci o stavu dané věci či člověka.

Při přepravě pacientů na urgentní oddělení je potřeba rychle zaznamenat a správně zpracovat potřebné dokumenty o pacientovi. K usnadnění této práce se prozatím využívá pouze psacích metod záznamů do dokumentací o pacientech. V nich se zapisuje například, která vyšetření pacient absolvoval nebo jaké bere léky. Důležité jsou i záznamy o nástrojích, které byly použity při zákrocích.

S tímto problémem by mohly pomoci NFC tagy. Po přečtení umožní získat informace, které léky pacient užívá, jaké přístroje byly použity při operaci, případně jakými metodami byl pacient vyšetřen. Díky tomuto bude zdravotní personál přehledně a rychle o všem důležitém informován.

Cílem této bakalářské práce je navrhnout a vytvořit aplikaci, jež správně načte informace z čárových či 2D maticový kódů nebo RFID tagu a ve správném rozlišení je zobrazí na mobilních přístrojích s operačním systémem Android. Dále musí aplikace umět propojit informace o entitách s NFC tagy s databází určenou pro uchovávání těchto informací.

Bakalářská práce po úvodní kapitole dále obsahuje kapitolu OS Android, která informuje o vývoji, architektuře, aplikaci a softwaru operačního systému Android. Třetí kapitola - Informační technologie obsahuje informace o čárových kódech a RFID systému. Ve čtvrté kapitole jsou informace o analýze, návrhu a implementaci aplikace. Pátá kapitola obsahuje uživatelskou dokumentaci a šestá programátorskou dokumentaci.

## 2. Operační systém Android

Mobilní aparáty použité při této bakalářské práci obsahují Operační systém Android (dále jen OS Android).

OS Android je jeden z nejvíce využívaných operačních systémů. Vytvořila jej společnost Google převážně pro uživatele PDA, tabletů a tzv. chytrých telefonů. Od základu byl sestaven pro snadné vytváření mobilních aplikací. Umožňuje snadný přístup do všech aplikací, které telefon nabízí, např. obsluha telefonních hovorů, posílání textových zpráv (SMS) nebo využívání fotoaparátů.[1]

### 2.1 Vývoj OS

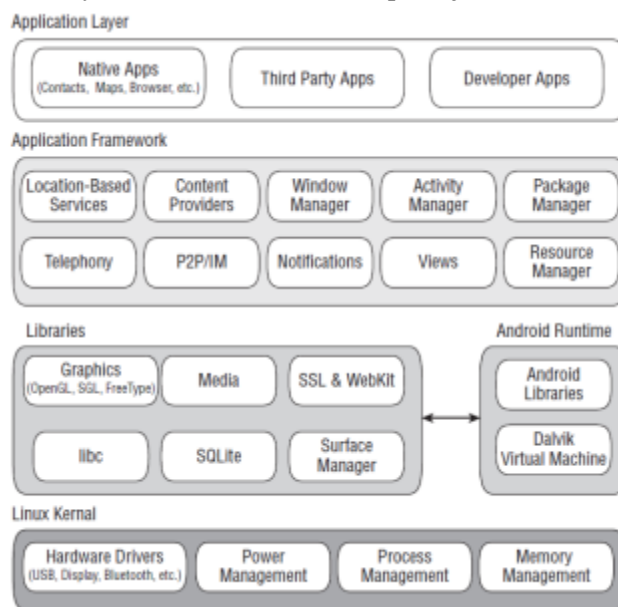
OS Android prošel během čtyř let několika změnami. Všechny jsou postaveny na Linuxovém jádře. První verze vyšla v roce 2008 – byla to verze Android 1.0. Dále byly vyvinuty verze:

- Android 1.1
- Android 1.5 (Cupcake)
- Android 1.6 (Donut)
- Android 2.0/2.1 (Eclair)
- Android 2.2 (Froyo)
- Android 2.3/2.4 (Gingerbread)
- Android 3.0/3.1/3.2 (Honeycomb)
- Android 4.0 - 4.0.4 (Ice Cream Sandwich)

Nejnovější verze je Android 4.1 – 4.2 (Jelly Bean), která se objevila na trhu v roce 2012.[9]

## 2.2 Architektura OS

Architektura OS Android se skládá z pěti vrstev (znázorněna na obrázku č. 1). Každá vrstva provádí různé operace a vystupují jako samostatné komponenty. Jelikož v praxi dochází ke vzájemné spolupráci jednotlivých částí, nelze tvrdit, že pracují zcela samostatně. [1]



Obrázek č. 1 Schéma architektury OS Android. Převzato z [2]

Jednotlivé části architektury OS Android tvoří:

- Linux Kernel (linuxové jádro) provádí implementaci abstrakce mezi použitým hardwarem a softwarem. Při spuštění zařízení je linuxové jádro zavedeno do operační paměti a předáno řízení. Díky tomu má neustálou kontrolu nad systémem a koordinuje činnosti všech běžících procesů (například souběžný běh aplikací), správa paměti, bezpečnost, správa sítě.[7]
- Libraries (knihovny) jsou napsány v jazyce C / C++ a využívají je různé části systému. Následuje soupis některých vybraných typů knihoven:
  - FreeType knihovna pro bitmapové a vektorové vykreslení písma.
  - Grafické knihovny, které zahrnují SGL a OpenGL pro 2D a 3D grafiku.
  - SQLite pro nativní podporu databáze.
  - SSL a WebKit pro podporující využití šifrovacího protokolu pro bezpečnou internetovou komunikaci.[7]

- Android Run Time (běhové prostředí Androidu), tato vrstva slouží pro běh aplikace. Obsahuje virtuální stroj DVM (Dalvik Virtual Machine), který vznikl k překladu aplikací, které jsou v jazyce java, pro jejich instalaci na různé typy systémů. Proto má DVM za úkol převést aplikace z kódu psaného na kód nativní. Vrstva obsahuje i standardní knihovny jazyku Java.[7]
- Application Framework (aplikační rámec) poskytuje přístup k různým službám, které mohou využívat aplikace v systému Android. Některé typy služeb:
  - View Systém umožňuje tvorbu graficko-uživatelského rozhraní a použití prvků, jako textové pole, seznamy, zaškrťovací políčka aj.
  - Notification Manager zobrazuje stavový řádek s vlastním upozorněním.
  - Content Providers slouží k přístupu a práci s obsahem jiných aplikací.
  - Activity Manager ovládá životní cyklus aplikací od startu, průběhu, až po ukončení.
  - Package Manager obsahuje informace o nahrených aplikacích do operačního systému.[7]
- Application Layer (aplikační vrstva) je vrstva která obsahuje všechny aplikace nativní a třetí strany. Ty jsou postaveny na aplikační vrstvě pomocí stejné API knihovny. Aplikační vrstva probíhá v běhovém prostředí Androidu pomocí tříd a služby zpřístupněné z aplikačního rámce. Jedná se o aplikace, které jsou předinstalované, ale i stažené aplikace z online katalogu (Google Play).[7]

## 2.3 Android aplikace

Každá aplikace se skládá z komponent. Níže jsou uvedeny základní komponenty pro tvorbu Android aplikace.

- Activity (aktivita)

Komponentu si lze představit, jako prezentační vrstvu aplikace. Aktivita poskytuje dialogové okno, prostřednictvím kterého se může komunikovat. V aplikaci se určuje jedna aktivita jako hlavní, která se zobrazí ihned po spuštění aplikace. Dále může pokračovat dalšími aktivitami, které se spouštějí za účelem nějaké funkce. Vždy když se spustí další aktivita, pozastaví se systém předchozí aktivity a ta je uložena do zásobníku. Zásobník funguje jako systém LIFO (last in, first out – poslední dovnitř, první ven). Mechanismy pro zachování zdrojů jsou nazývány životní cyklus aktivity.[7]

Životní cyklus aktivit se skládá ze tří smyček:

1. Úplný životní cyklus aktivity – odehrává se mezi voláním metody onCreate (vytvoření aktivity) a metody onDestroy (zrušení aktivity).
2. Viditelný životní cyklus aktivity – odehrává se mezi voláním metody onStart (spuštění aktivity) a metody onStop (zastavení aktivity).
3. Životní cyklus aktivity v popředí – odehrává se mezi voláním metody onResume (pokračování běhu aktivity) a metody onPause (pozastavení aktivity).

- Services (služby)

Poskytují dlouhotrvající operace na pozadí, ale neposkytují graficko-uživatelské rozhraní. Spouštějí se mohou komponentou aplikace. Služby budou stále aktivní na pozadí i při přejití do jiné aplikace. Stejně jako aktivity, mají životní cyklus i služby.

Životní cyklus služeb dělíme na dva typy:[10]

1. Started služba – spouštěna metodou startService () a ukončena metodou stopService ().
2. Bound služba – spouštěna metodou bindService () a ukončena metodou unbindService ().

- Content Providers (Poskytovatelé obsahu)

Umožňuje zpřístupnit data, se kterými je pak možnost pracovat. Načítat, ukládat data a tím je zpřístupnit pro všechny aplikace. Podle rozsahu oprávnění se rozlišuje, kdo má možnost data pouze číst a kdo má právo data i zapisovat.[7]

- Intents (záměry)

Je to abstrakce operace, kterou je potřeba vykonat. Celý tento prostor se skládá z komponent (aktivity) a ze zpráv mezi komponentami (záměry).

Záměr se skládá ze tří částí:

- Z činnosti, která se má vykonat
- Z parametru, nad kterým má být tato činnost vykonána
- Z aplikace, která má tuto akci provést

Mezi příklady činností, které jsou považovány za záměr, patří odeslání SMS zprávy, zobrazení webové stránky nebo příjem hovoru.[7]

- **Broadcast Receivers (Přijímače)**

Slouží k naslouchání oznámení o stavu z vnějšku i zevnitř aplikace. Přijímač nemá stejně jako služby uživatelské rozhraní. Příkladem použití přijímače patří reakce na oznámení na příchozí SMS zprávy nebo oznámení o nízkém stavu baterie.[7]

## **2.4 Software Development Kit (SDK)**

Jde o balíček, který obsahuje vývojové nástroje pro tvorbu aplikace určitých operačních systémů, hardware platform nebo herních konzol. SDK obsahuje dokumentaci atd. Android SDK obsahuje totožné dokumenty, knihovny, jako jiné SDK. Navíc obsahuje Android SDK knihovny, které lze využít pro tvorbu výkonných mobilní aplikací pro OS Android.[7]

### **2.4.1 Emulátor**

Android SDK také obsahuje virtuální mobilní zařízení, které se nazývá emulátor. Jde o nástroj, na kterém lze testovat a ladit aplikace. Jde o implementaci virtuálního stroje Dalvik, který vytváří platnou platformu pro provozování Android aplikací, jako by se jednalo o fyzické zařízení s OS Android. K dispozici je řada uživatelských rozhraní pro prezentaci různých konfigurací hardwaru, každá je s různou velikostí obrazovky, rozlišením, orientací a hardwarových funkcí a tak lze vytvářet různá mobilní zařízení. Díky plnému připojení se lze na emulátoru připojit k internetu a také se může simulovat příchozí a odchozí hovory nebo SMS zprávy.[7]

### 3. Identifikační technologie

V této bakalářské práci jsou využity QR kódy (viz Obsah CD - Evidence přístrojů.pdf), které se dají vygenerovat na několika internetových stránkách, ovšem je možné použít i jiné typy. Popsány jsou v této třetí kapitole.

V dnešní době se identifikační technologie využívají v mnoha různých odvětvích. Snahou je, co nejrychleji a efektivně přenést určitou informaci. Ve zdravotnictví se využívá zejména systém GS1, který se zabývá tvorbou mezinárodních, ale i mezioborových logických a dodavatelsko-odběratelských řetězců.[6]

Mezi nejvyužívanější typy identifikačních technologií, které se dnes používají patří QR kódy, dále také EAN kódy nebo RFID tagy.

#### 3.1 Čárové kódy

Jsou jedny z nejrozšířenějších prostředků identifikačních technologií. Jsou tvořeny černotiskem ve tvaru pruhů nebo mozaiky. Černé čáry jsou od sebe definovanou šířkou odděleny, díky čemuž jsou informace jedinečné. Informaci nám nenesou pouze mezery mezi čarami, ale také tištěná čára. Mezi základní prvky čárových kódů patří nejenom šířka modulu (nejužší mezera), délka kódu (snímání začátku kódu – Start a poslední značku kódu – Stop), ale také kontrast (rozdíl jasů odrazu pozadí a čáry ku odrazu čáry k jasů odrazu pozadí).[11]

Typy čárových kódů dělíme na jednodimenzionální, dvoudimenzionální a třídimenzionální kódy.

##### 3.1.1 Jednodimenzionální kódy

Také často značené 1D kódy. Mají omezenou kapacitu, proto nejčastěji kódují numerické nebo alfanumerické řetězce. Toto kódování se nejvíce využívá k identifikaci předmětu nebo externí databáze. Mezi jednodimenzionální kódy patří např. Code 128, Code 39, Codabar nebo dále také EAN kódy 8 a 13. [12]



### 3.1.1.1 EAN kód

Nejpoužívanější EAN ( European Article Number) kód je kód EAN-13 (ukázka na obrázku č. 2). Tento kód je vícemístný a je v dnešní době nejvíce využíván k označování výrobků. EAN-13 kód je schopný v sobě uchovat kódy ISBN knížek nebo ISSN časopisů.[11]

U EAN-13 jsou jednotlivé symboly zakódovány do 13-ti čísel, která jsou rozdělena na čtyři části:

- První část – první dvě nebo tři číslice identifikují zemi zaregistrovaného výrobce, pokud ale vznikl z knižní podoby ISBN, obsahuje první tři číslice čísla 978 nebo 979. Pokud vznikl z podoby časopisu ISSN, tak jsou první tři čísla 977.
- Druhá část – je kód výrobce, v závislosti na systému kódování se skládají ze čtyř nebo pěti číslic.
- Třetí část – je kód výrobku. Ten se skládá z pěti číslic.
- Čtvrtá část – je kontrolní část, která se odpočítá z modulu 10, tzv. samodetekující kód.[11]



Obrázek č. 2 Ukázka EAN-13 kódu. Převzato z [11]

### 3.1.2 Dvoudimenzionální kódy

Oproti jednodimenzionálním kódům byly dvoudimenzionální kódy vytvořeny pro uložení většího množství dat, protože mají vyšší kapacitu. Poprvé byly použity v lékařském a elektrotechnickém průmyslu. Obvykle obsahují informace o daném předmětu, které jsou uloženy ve tvaru matice. Podle způsobu uložení se dělí 2D kódy na:

- Skládané (Stacked) a více řádkové, vznikly z jednodimenzionálních kódů. Jsou tvořeny z čar a mezer proměnné šíře. Patří sem např. kódy Codablock, kód PDF 417.
- Maticové kódy (Matrix code) jsou tvořeny dvourozměrnými souřadnicemi tmavých bodů v matici, tyto body mají stejné rozměry. Kód může mít různé tvary, šestiúhelník, kruh nebo čtverec. Mezi maticové kódy patří např. Data Matrix nebo QR Code.
- Ordinální čárové kódy, ty jsou vertikálně redundantní, díky tomu se může výška sloupců změnit, aniž by byla ztracena jakákoliv data.
- Dvoudimenzionální kódy, ty nesou informace, jak ve svislém směru, tak v horizontálním.

V dnešní době se nejvíce využívají maticové kódy a z nich se můžeme nejvíce setkat s QR kódy, (viz obrázek 3).[13]

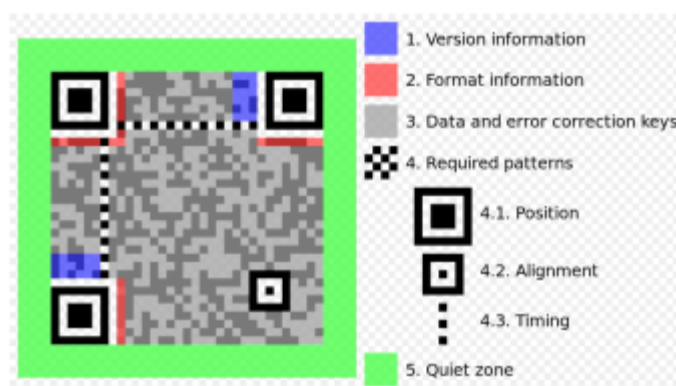
#### 3.1.2.1 QR kód

QR Code (Quick Response Code), byl vytvořen japonskou společností Denso-Wave v roce 1994. QR kódy jsou striktně dané podle normy ISO 18004:2006.[14]

Každý QR kód je tvořen čtverci, tzv. Finders, které se nacházejí v rozích kódu. Dále mezi Finders jsou řádky a sloupce vyhrazené na tzv. Timing vzory, což je střídání černých a bílých bodů. V neposlední řadě je na QR kódu vidět tzv. Alignments, což jsou čtverce menšího tvaru než Finders, lišící se od nich černou tečkou ve čtverci.[15]

QR kód se skládá z funkčních elementů (viz obrázek č. 3):

1. Informace o verzi.  
Je to informace o použité velikosti.
2. Informace o formátu.  
Podle rozpoznáního textu pozná, zda se jedná o webovou stránku nebo o telefonní číslo.
3. Data a samoopravné kódy.
4. Povinné vzory.  
Jsou terče pro správné zaměření.
5. Tichá zóna.  
Je ohraničení celého kódu bílou barvou. Stejně, jako povinné vzory slouží ke správnému zaměření.



Obrázek č. 3 Ukázka QR kódu. Převzato z [15]

### 3.1.3 Třídímenzionální kódy

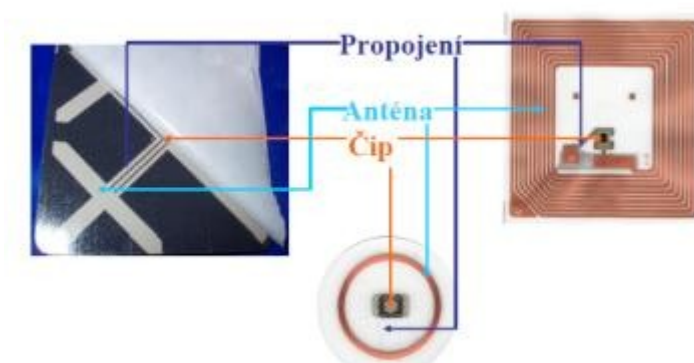
Také nazývané Bumpy Barcode. Jde o typické čárové kódy, pouze se liší ve způsobu tisku a snímání. Jsou totiž tištěny hloubkově neboli embosovány, což můžeme např. vidět u platebních karet do bankomatu. Snímání se pak provádí podle výškových změn a rozdílů. [13]

## 3.2 Knihovna Zxing

Knihovnu Zxing vytvořila společnost Google. Je psaná v jazyku Java pro čtení a zápis 1D a 2D čárových kódů a je schopna rozpoznat kód bez komunikace se serverem. Také je nabízena v jiných portech, jako např. C nebo C++.

## 3.3 RFID

RFID tag (ukázka na obrázku č. 4) slouží podobně jako čárové kódy k identifikaci, ale pracuje na principu radiofrekvenčního systému pomocí elektromagnetických vln. Skládá se z antény, čipu a propojení mezi nimi.



Obrázek č. 4 Ukázka RFID tagů. Převzato z [17]

Mezi hlavní výhody RFID oproti čárovým kódům patří:

- Možnost identifikace přes různé materiály.
- Možnost identifikovat více tagů najednou.
- Snímání ze vzdálenosti 0 – 15 m, oproti tomu čárové kódy nemohou být vzdáleny víc, jak 1 m.
- Rychlost čtení.
- K přenosu dat dochází i přes nepříznivé optické nebo atmosférické podmínky.[16],[17]

### **3.3.1 Komponenty RFID systému**

Do RFID systému patří tři základní komponenty.[16]

1. RFID tag – transportér.
2. RFID reader – čtecí zařízení, které může být buď čtecí bránou nebo různým ručním čtecím zařízením.
3. Middleware, řídicí software.

### **3.3.2 Rozdělení RFID systému podle frekvenčního pásma**

Systém RFID využívá radiových vln, které podle typu vlnové délky procházejí různými materiály. Podle typu kmitočtového pásma lze dosáhnout na čtečku i přes nepříznivé podmínky okolního prostředí. Ovšem závisí na volbě frekvence, dle které lze rozdělit RFID systém na čtyři frekvenční pásma.[16],[17]

1. Low Frequency (Nízké frekvence) – frekvence 125 – 134 kHz, dosah čtecí vzdálenosti cca. 20 cm a přenosová rychlost je nízká.
2. High Frequency (Vysoká frekvence) – frekvence 13,56 MHz, dosah čtecí vzdálenosti cca. 1 m a má dostatečnou přenosovou rychlost.
3. Ultra High Frequency (Velmi vysoká frekvence) – frekvence 860 – 960 MHz, přenos informací zvládá do 3 m a přenosová rychlost je vysoká.
4. Microwave (Mikrovlnná frekvence) – frekvence 2,54 GHz nebo 5,8 GHz, přenos zvládá oproti ostatním frekvencím i přes kov, prach nebo kapalinu a má velmi vysokou přenosovou rychlost.

### **3.3.3 Rozdělení RFID tagů**

Podle typu napájení dělíme RFID tagy na aktivní a pasivní.

#### **Aktivní tagy**

Mají vlastní zdroj energie, který využívají na napájení integrovaného čipu anebo k posílení signálu. Díky tomu jsou schopny vysílat s vyšším výkonem, a proto jsou vhodné do rušivého prostředí. Mohou obsahovat snímače k měření fyzikální veličiny. Jejich nevýhodou je složitá konstrukce, a proto jsou na trhu dražší. Dále je také nevýhodou jejich menší životnost kvůli kapacitě baterie.[16]

#### **Pasivní tagy**

Energii přijímají z antény, tudíž nepotřebují zdroj napájení. Oproti aktivnímu tagu mají výhodu v levné údržbě, cenově jsou levnější a mají dlouhou životnost.[16]

## 4. Analýza, návrh a implementace aplikace

V následující kapitole je umístěn popis základních požadavků, jejich analýza, návrh řešení a implementace aplikace.

### 4.1 Základní požadavky

Základním požadavkem bylo vytvoření aplikace pro mobilní zařízení, která bude umožňovat identifikaci a čas příjmu pacienta na základě kódu sejmутého aplikací. Dále bude moci přiřazovat léky a přístroje k tomuto pacientovi s daným časem akce, která bude identifikována opět pomocí naskenovaného kódu. Posledním požadavkem bylo vytvoření i přehledného souhrnu pacientů, léků, vyšetření a přístrojů v databázi.

### 4.2 Datová analýza

Data aplikace jsou uložena v rámci databáze, která nebyla součástí této práce. Proto v rámci datové analýzy bylo pouze seznámení se s řešením databáze, jež byla následně použita. Diagram je obsažen v příloze [I]. Jeho řešení obsahovalo 23 tabulek, z nichž byly vybrány následující:

- Pacienti – obsahuje údaje o jednotlivých pacientech, pro aplikaci jsou důležité atributy: kód, jméno, příjmení.
- Leky\_ZIP – obsahuje údaje o jednotlivých lécích, u této entity se informuje o attributech: pořadí, název a kód léku.
- Pristroje – obsahuje údaje o jednotlivých přístrojích, aplikace bude pracovat s attributech: pořadí, název a kód přístroje.
- Vysetreni\_ZIP – je vazební tabulkou mezi entitami pacient a přístroj, obsahuje atributy: pořadí, název a kód vyšetření.
- Pacienti\_has\_leky – je vazební tabulkou mezi entitami pacient a lék, obsahuje atributy: datum (přesné datum a čas podání léku pacientovi) a id pacienta a léku.
- Pacienti\_has\_pristroje – je vazební tabulka mezi entitami pacient a přístroj, obsahuje atributy: datum (přesné datum a čas použitého přístroje na pacientovi) a id pacienta a přístroje.

- Pacienti\_has\_vysetreni – je vazební tabulka mezi entitami pacient a vyšetření, obsahuje atributy: datum (přesné datum a čas vyšetření pacienta).

### 4.3 Funkční analýza

Hlavními funkcemi aplikace jsou:

- Přehled pacientů, léků, vyšetření a přístrojů uložených v databázi.
- Identifikace pacienta a následné přiřazení léků a vyšetření či přístrojů na něm použitých.

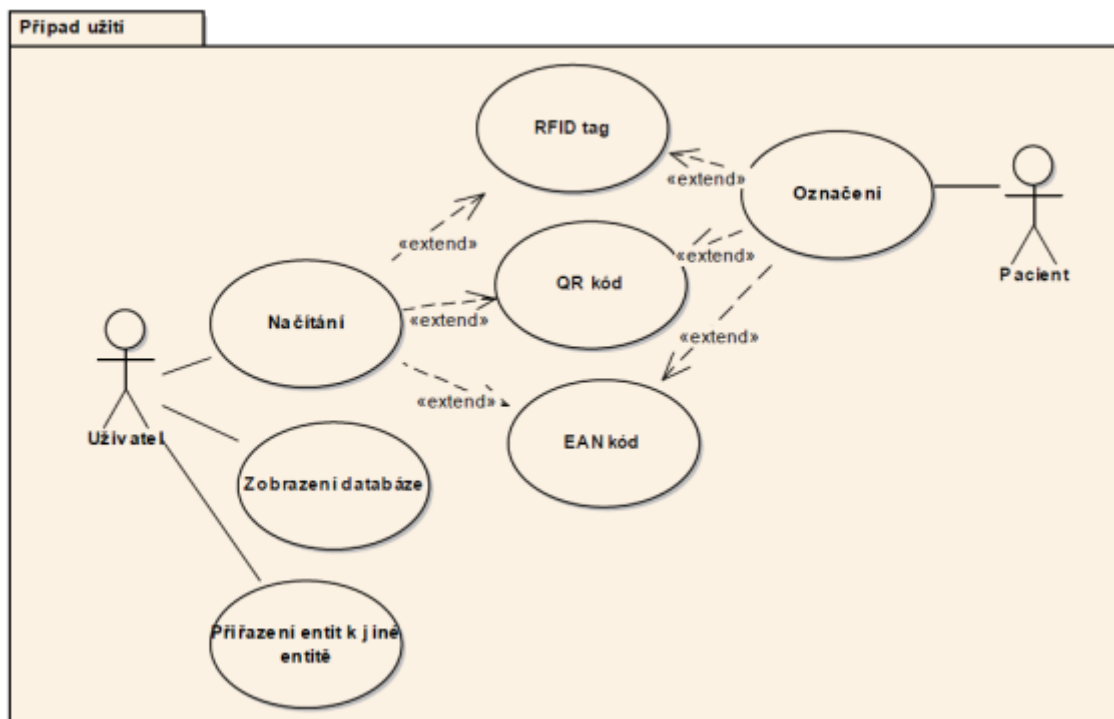
První funkce zobrazuje všechny záznamy zvolené entity, tedy jejich jména (případně názvy).

Druhou funkci lze rozdělit na několik částí, které jsem řešila samostatně a to:

1. Načtení čárového a QR kódu.
2. Načtení informací o dané entitě z databáze.
3. Zobrazení informací o dané entitě.
4. Přiřazení entity k jiné entitě (vlození záznamu do databáze).
5. Načtení NFC tagu.

Veškeré funkce jsou vidět na obrázku č. 5, který představuje use case diagram aplikace.





Obrázek č. 5 Diagram případů užití

V první části je třeba vytvořit metodu pro čtení čárových a QR kódů, z nichž získáme identifikátor entity (pacienta, léku či přístroje). Tento identifikátor pak použijeme v druhé části, kdy načteme potřebné informace z databáze. Ve třetí části zobrazíme údaje o dané entitě, kdy zobrazujeme pouze zlomek z možných informací z kapacitních důvodů. Následně, pokud načítaná entita byla lék či přístroj, přiřadíme tento záznam k pacientovi, který byl identifikován dříve. Přiřazení probíhá zápisem do databáze i s aktuálním časem. Poslední možností je načtení NFC tagu, který může být nositelem identifikátoru určitého pacienta. Po identifikaci kódu pacienta je dána možnost přiřadit mu lék, vyšetření nebo přístroj, které byly aplikovány či použity.

Veškeré zápisy do databáze budou vkládány s tzv. časovou značkou, aby byla možná zpětně dohledat doba zápisu.

V průběhu může dojít i k několika nežádoucím situacím, které je třeba ošetřit. Patří sem například nestabilita aplikace, neplynulost údajů či chybějící údaje v databázi, nedostupnost připojení k internetu a další. Vše je třeba vyřešit minimálně upozorněním na daný stav nebo situaci uživatele.

## 4.4 Návrh uživatelského prostředí

Uživatel po spuštění aplikace má možnost výběru načtení čárového kódu pomocí fotoaparátu a nebo identifikaci tagu přiložením RFID k mobilnímu zařízení. Dále má možnost zobrazení evidované databáze.

Pro přehledný popis aplikace byly vytvořeny diagramy aplikace InformaceNFC. Tyto diagramy napomohly alespoň k napodobitelné realitě a ke snaze vytvoření využitelné aplikace. Diagramy jsou znázorněny na obrázcích č. 6 a 7.

Na obrázku č. 6 je zobrazeno hlavní menu, napravo od jednotlivých tlačítek je popis jejich funkcí.



Obrázek č. 6 Diagram aplikace Informace NFC, hlavní menu

Na obrázku č. 7 je znázorněn návrh pro zobrazení načtených kódů pacienta a typu výkonu, který byl použit. Jednotlivé funkce zobrazení a tlačítka jsou v obrázku popsány napravo.

		Zobrazení kódu pacienta.
		Zobrazení typ kódu pacienta.
		Čas podání léků nebo čas vyšetření.
		Výpis kódu podaného léku nebo vyšetření.
		Typ kódu.
Zpět do menu	Uložit	Tlačítko zpět do menu, vrátí aplikaci na stránku hlavní menu. Tlačítko uložit, uloží záznam do databáze.
Vyšetření		Po stisknutí jednoho ze tří tlačítek se spustí skaner, který vyfotí kód a přiřadí k pacientovi daný výkon.
Léky		
Přístroje		

Obrázek č. 7 Návrh pro zobrazení načtených kódů

## 4.5 Návrh řešení

Tato aplikace byla navržena zejména k ušetření času pro daný personál na urgentním oddělení. Protože jde na tomto oddělení nejvíce o život ohrožující stavy, není v daných chvílích čas k zápisu do dokumentací. Z tohoto důvodu byla navržena tato aplikace, která ušetří pár minut drahocenného času a díky tomu by pak bylo možné rychleji reagovat na zdravotní stav pacientů.

Pacient by byl označen RFID systémem nebo čárovým (QR) kódem, který by sděloval jeho identifikaci a čas. Tato identifikace by se sejmula prostřednictvím mobilu či tabletu, po spuštění aplikace a výběru snímání. Po zjištění identifikace je dále možné pokračovat v zakládání informací. Pomocí snímání se vloží určité druhy vyšetření, aplikování léků nebo použité přístroje do databáze. Tímto způsobem má zdravotnický personál možnost kontrolovat či později dohledat informace o daném pacientovi.

Aplikace je použitelná i na jiných odděleních, zejména pak ke sledování použitých předmětů nebo nástrojů, které byly použity na sálech a musejí být sterilizovány. Tím by se dalo zamezit jejich ztrátám. Další možností je využití aplikace na operačních sálech, kde se nachází různé sadové boxy s předměty k operacím. Pomocí skenování by sestra zjistila, co je obsahem boxu, aniž by jej musela rozbíjet. Vše by probíhalo sterilně. Podobné aplikace jsou již využívány ke kontrole léků, kde se

vždy při dodávce nových léčiv zaznamená jejich kód s počtem převzatých balení. Po určitém počtu využitých léčiv dojde k informativnímu volání, které upozorní, že se snížil počet léků na skladě a je zapotřebí objednat nové.

## **4.6 Implementace**

Pro tvorbu aplikace bylo použito prostředí Eclipse s pluginem ADT. Dále byla aplikace testována na mobilním zařízení LG Optimus One P500, Samsung Google Nexus S i9023 a emulátoru.

Aplikace byla navržena pro urgentní oddělení. Po několika osobních návštěvách tohoto oddělení a schůzkách s personálem byla navržena aplikace, která by splňovala jejich požadavky jako budoucích uživatelů.

Vzhledem k nemožnosti přístupu do informačního systému fakultní nemocnice bylo nutné vytvořit vlastní systém. Toto je také jeden z důvodů, proč aplikace nebyla otestována přímo v provozu. Zařízení bylo proto testováno pouze v domácím prostředí s mobilními zařízeními a pomocí kódů, které byly zpřístupněny.

## 5. Uživatelská dokumentace

Obsahem této kapitoly je uživatelská dokumentace aplikace popisující její základní funkce.

### 5.1. Aktivní aplikace

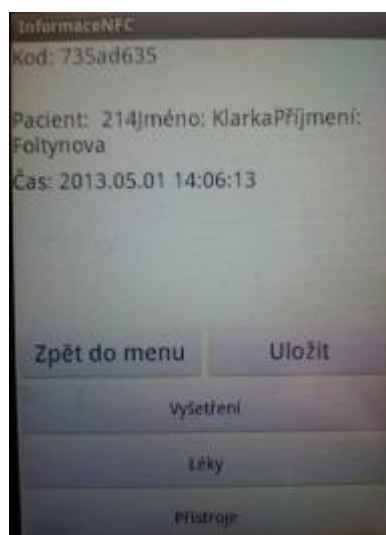
Po nainstalování a spuštění (viz obrázek č. 8) je možno seznámit se s aplikací InformaceNFC prostřednictvím *nápovědy*. Ta přesně popisuje, které služby aplikace nabízí (např. načtení evidenční databáze, postup snímání kódů a RFID tagů a ukládání evidence do databáze) a v neposlední řadě obsahuje kontakt na autora.



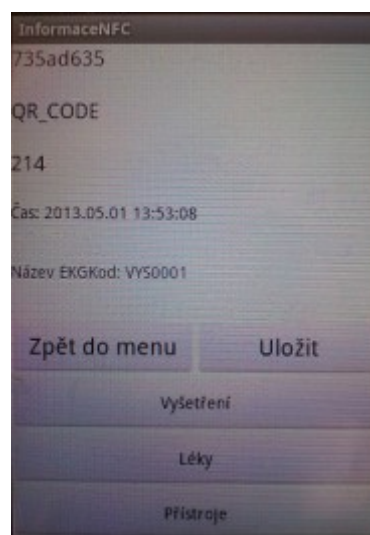
Obrázek č. 8 Spuštěná aplikace Informace NFC

## 5.2. Scan a NFC systém

Po seznámení s nápovědou lze načíst data podle typu tagu. Pokud bude pacient označen QR kódem nebo ean kódem, vybere se nabídka *Scan*, pomocí které se naskenuje kód pacienta. Poté, co se zobrazí kód pacienta, je zapotřebí k zobrazení jeho jména uložit čas příjmu. Čas příjmu se uloží pomocí tlačítka *Uložit* do databáze, prostřednictvím kódu pacienta. V nabídce pak stisknutím dalšího tlačítka z výběru *Vyšetření, Léky, Přístroje*, bude proveden záznam do jeho databáze dle výběru, zda jde o vyšetření, přístroj nebo lék, který se použije či aplikuje na pacientovi. Tento záznam bude uložen i s údajem o času opět pomocí tlačítka *Uložit*. Výběrem tlačítka *NFC*, bude možné přečtení informace z tagu. Na obrázcích č. 9 a 10 lze vidět zobrazený kód pacienta a léku po naskenování.



Obrázek č. 9 Po naskenování pacienta



Obrázek č. 10 Po naskenování léku

Obrázek č. 11 zobrazuje záznam z databáze (id\_záznamu, doba načtení, id\_pacienta a id\_vysetreni).

	 Upravit	 Kopírovat	 Odstranit	11	2013-05-01 13:52:47	412	1
---	---	---	---	----	---------------------	-----	---

Obrázek č. 11 Záznam z databáze

### 5.3. Databáze

Poslední možností výběru je tlačítko *Databáze*, ve které lze zobrazit tři nejdůležitější databáze a to *Pacienti*, kde je vypisováno jméno a příjmení pacienta, dále tlačítko *Léky*, kde je zaznamenán název léku a čas podání a tlačítko *Přístroje*, kde je vypsáno pořadí a název přístroje. Na obrázcích č. 12 a 13 je zobrazena spuštěná aplikace databáze *Pacienti* před výběrem zobrazení databáze a po zobrazení databáze.



Obrázek č. 12 Spuštěná aplikace Databáze

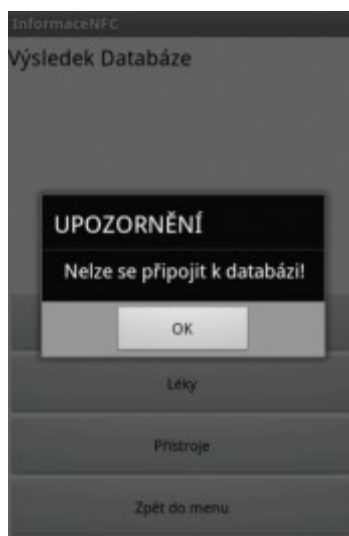


Obrázek č. 13 Po spuštění databáze Pacienti

Tlačítko *Zpět do menu*, jež je obsaženo i při skenování, umožňuje návrat do spuštěné aplikace.

## 5.4. Upozornění

V celé aplikaci může dojít k různým nežádoucím situacím. Proto pokud se objeví okno s upozorněním, došlo k určitému problému. O jaký typ problému jde se vždy zobrazí v samostatném okně. Např. pokud nedošlo k připojení wifi přijde hlášení „Nelze se připojit k databázi!“, jak je vyobrazeno na obrázku č. 14.



Obrázek č. 14 Okno s upozorněním



## 6. Programátorská dokumentace

V této kapitole je popsána struktura projektu a popis jednotlivých tříd. Zdrojové kódy jsou zpřístupněné v příloze na CD.

### 6.1 Struktura aplikace

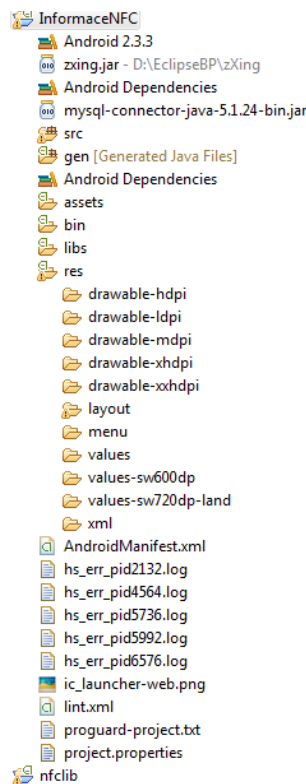
Aplikace se skládá z jednotlivých adresářů:

1. **src/** je složka obsahující zdrojové kódy aplikace v jazyce Java, třídy jsou rozděleny do balíčků.
2. **gen/** je složka obsahující zdrojové soubory jazyka Java, které jsou vygenerovány a lze je použít ve zdrojovém kódu.
3. **assets/** je složka obsahující externí složky, které jsou využívány, jako např. obrázek nebo textové řetězce.
4. **bin/** je složka obsahující archiv, který má příponu apk (Application Package File). Tento archiv obsahuje zkompilevanou aplikaci dex (Dalvik executable) a data pro správné zobrazení a běh aplikace.
5. **libs/** je složka se zkompilevanými knihovnami, jako např. BarcodeScanner.
6. **res/** tato složka obsahuje externí data, která jsou obsažena v aplikaci, jako např. GUI (graficko-uživatelské rozhraní), textové informace atd.
7. **AndroidManifest.xml** je soubor obsahující název aplikace, verzi aplikaci a všechny použité komponenty, jako jsou např. aktivity, služby atd.

### 6.1.1 Struktura adresáře

Adresář `res/` obsahuje externí data, která jsou rozdělena podle typu a konfigurace. Na začátku nové aplikace se automaticky v adresáři vytvoří některé podadresáře (viz obrázek č. 15). Tento adresář obsahuje složky:

- **drawable-hdpi**  
Složka obsahující ikony a obrázky aplikace s vysokým rozlišením.
- **drawable-ldpi**  
Tato složka obsahuje ikony a obrázky aplikace s nízkým rozlišením.
- **drawable-mdpi**  
Tato složka obsahuje ikony a obrázky aplikace se středním rozlišením.
- **drawable-xhdpi**  
Složka obsahující ikony a obrázky aplikace s velmi vysokým rozlišením.
- **drawable-xxhdpi**  
Tato složka obsahuje ikony a obrázky aplikace s extra velmi vysokým rozlišením.



Obrázek č. 15 Adresář `res`

- **layout/**

Obsahuje rozložení uživatelských rozhraní , která jsou použita v jednotlivých aktivitách a dialogů. Složka je uložena jako soubor XML.

- activity\_main.xml

Obsahuje rozložení tlačítka *Nápověda*, *Scan*, *Databáze* a *NFC*.

- help.xml

Obsahuje rozložení kód k zobrazení Nápovědy ze složky Values souboru Strings.xml.

- mysql.xml

Obsahuje dva Textviewy k zobrazení Databáze, po výběru jednoho ze tří tlačítek, Pacienti, Léky, Přístroje. Dále obsahuje tlačítko *Zpět do menu*.

- nfc.xml

Obsahuje Textview k zobrazení kódu pacienta.

- scan.xml

Obsahuje pět Textviewy k zobrazení načtených kódů a času. Tři tlačítka (Přístroje, Léky, Vyšetření) slouží k načtení vybrané akce, dále obsahuje tlačítko k ukládání a k navrácení zpátky do menu.

- **menu/**

Obsahuje XML soubor s definovaným menu.

- **values/**

Tato složka obsahuje XML soubory a to:

- Dimens.xml

Obsahuje rozměry, které jsou definovány pro aplikaci.

- Strings.xml

Obsahuje texty v souborech xml, které jsou mimo zdrojový kód a jsou tak přehlednější.

- Styles.xml

Obsahuje nadefinované kompatibility, které se zobrazí, jako je např. barva nebo velikost písma.

- **xml/**

Obsahuje soubor nfc\_filtr.xml, který definuje typy tagů možných ke čtení.

## 6.2 Struktura tříd

Pro orientaci v kódech jsou rozepsány hlavní aktivity, použité funkce a výjimky v nich.

### 6.2.1 Hlavní aktivity, třídy

Aktivity jsou nejdůležitější soubory. Obsahují nejenom zdrojové kódy, ale také informují o uživatelském rozhraní. Jsou jednotlivými obrazovkami, se kterými je pracováno. Aktivity jsou potomky třídy `android.app.Activity`.

Tato aplikace obsahuje tři aktivity:

- **MainActivity**  
V prostředí `MainActivity` je použita metoda `onCreate()`, která se aktivuje po spuštění aktivity. V té jsou vytvořeny objekty `View`, které jsou jednotlivě spouštěny funkcí `setContentView(R.layout.activity_main)`. Po vybrání funkce tlačítka se provede akce.  
Např. `btnScan = (Button) findViewById(R.id.btnScan);`  
`setListeners();`  
Dále je použita metoda volání třídy, která je řešena objektem `Intent`. Volán explicitní metodou je třída `Help`, která je popsána níže.
- **Help**  
Ve třídě `Help` je opět využita metoda `onCreate()`, která aktivuje spuštění `layout.help`, ten odkazuje na složku `values.String.xml`, kde se nachází text s nápovědou.
- **MySQL**  
Ve třídě `MySQL` pomocí metody `getConnection()` dochází k připojení k databázi a po té jsou provedeny zadané příkazy.

### 6.2.2 Použité metody a funkce

Ve třídách byly použity kromě už zmíněných metod `onCreate()`, `getConnection()`, také další metody, jako např. `startActivityResult()`, která spustí aktivitu `onActivityResult()`. Protože se v kódu nachází více aktivit, jsou jednotlivé `onActivityResult()` od sebe rozlišeny `requestCode` nesoucích určitou hodnotu. Výše zmíněná metoda `setOnClickListener` nastaví po stisknutí tlačítka vybraný `layout`.

### **6.2.3 Výjimky**

Ve využitých výjimkách byl použit `AlertDialog.Bulder`, který otevírá okno s doplňujícími informacemi. Toto okno umožňuje vepsat hlavní titulek, odpovídající zprávu k události a potvrzovací tlačítko *OK* o přečtení zprávy.

## 7. Závěr

Úkolem této práce bylo vytvořit mobilní aplikaci, která bude schopna načíst data z čárových kódů nebo 2D maticových kódů a RFID tagů, zobrazit a uložit je do databáze.

V průběhu zpracování došlo k problému s databází, který se nakonec podařilo vyřešit. Databáze je zobrazitelná a lze do ní ukládat veškeré záznamy. V uložených záznamech jsou obsaženy informace o čase příjmu pacienta, druhu vyšetření, čase vyšetření, aplikovaných léčích, čase aplikování, použitých přístrojích a čase použití. Funkce a jejich implementace je popsána v kapitolách 6 a 7.

Použity jsou QR kódy, které jsou vloženy v příloze [II].

Aplikaci InformaceNFC je možné využít k vytváření mobilních aplikací, které obsahují funkci pro snímání pomocí EAN, QR kódu a nebo RFID tagu. Například je možné sledovat použité nástroje na operačních sálech kdy, kde a jakým způsobem byly využity. Tak budou mít nejenom personál, ale i všechna oddělení přehled o pracovních nástrojích a bude možné předcházet jejich ztrátě. Dalším pracovištěm, kde by bylo možné využití těchto kódů, jsou lékárny, kde by se pomocí databáze sledovala expirace léčiv.

Jedná se o perspektivní aplikaci, jejíž využití je velmi rozsáhlé a pevně věřím, že v budoucnu významně přispěje ke zvýšení efektivity práce ve zdravotnictví.

## Literatura

- [1] MURPHY, Mark L. *Android 2: průvodce programováním mobilních aplikací*. Vyd. 1. Brno: Computer Press, 2011, 375 s. Průvodce (Grada). ISBN 978-80-251-3194-7.
- [2] MEIER, Reto. *Professional Android Application Development*. Indianapolis(USA): Wiley Publishing Inc., 2009. ISBN 978-0-470-34471-2. Dostupné na World Wide Web: <[http://www.google.cz/url?sa=t&rct=j&q=professional%20android%20application%20development&source=web&cd=2&ved=0CEAQFjAB&url=http%3A%2F%2Fandroidteam.googlecode.com%2Ffiles%2Fprofessional\\_android\\_application\\_development.pdf&ei=GUFT7DyD8\\_IswaXqumADw&usg=AFQjCNFyzLrx89xFrFq3rvWtSjPohcvTQA&sig2=ynMy9esnk8kst9B5kCTcKg](http://www.google.cz/url?sa=t&rct=j&q=professional%20android%20application%20development&source=web&cd=2&ved=0CEAQFjAB&url=http%3A%2F%2Fandroidteam.googlecode.com%2Ffiles%2Fprofessional_android_application_development.pdf&ei=GUFT7DyD8_IswaXqumADw&usg=AFQjCNFyzLrx89xFrFq3rvWtSjPohcvTQA&sig2=ynMy9esnk8kst9B5kCTcKg)>
- [3] ECKEL, Bruce. *Myslím v jazyku Java: knihovna programátora*. Praha: Grada Publishing, 2001. ISBN 80-247-9010-6.
- [4] WIKIPEDIA, the Free Encyclopedia. *NFC, Near Field Communication*. [online] [cit. 2012-10-16]. Dostupné na World Wide Web: <[http://en.wikipedia.org/wiki/Near\\_field\\_communication](http://en.wikipedia.org/wiki/Near_field_communication)>
- [5] ANDROID DEVELOPERS. The Developer's Guide [online][cit. 2012-10-12]. Dostupné na World Wide Web: <<http://developer.android.com/guide/index.html>>
- [6] CESKO.GS1 CZECH REPUBLIC. *GS1 Czech Republic*. Publikace vytvořeny organizací GS1 [online], c2012. Dostupné na World Wide Web: <<http://www.gs1cz.org/sluzby/>>
- [7] UJBÁNYAI, Miroslav. *Programujeme pro Android*. Vyd. 1. Praha: Grada, 2012, 187 s. Průvodce (Grada). ISBN 978-80-247-3995-3.
- [8] KONEČNÝ, Matěj. *Zdroják.cz: Dej Androidu tablety!* [online]. [cit. 2012-10-12]. Dostupné na World Wide Web: <<http://www.zdrojak.cz/clanky/dej-androidu-tablety/>>
- [9] WIKIPEDIA, otevřená encyklopedie. *Android\_(operační\_systém)* [online]. Datum poslední revize 29.11.2012 [cit. 2012-12-05]. Dostupné na World Wide Web: <[http://cs.wikipedia.org/wiki/Android\\_%28opera%C4%8Dn%C3%AD\\_syst%C3%A9m%29](http://cs.wikipedia.org/wiki/Android_%28opera%C4%8Dn%C3%AD_syst%C3%A9m%29)>
- [10] KYPTA, Tomáš. *Vyvíjíme pro Android* [online]. Datum publikování 16. 3. 2011 [cit. 2013-01-04]. Dostupné na World Wide Web: <<http://www.abclinuxu.cz/clanky/vyvijime-pro-android-uvod>>

- [11] WIKIPEDIA, otevřená encyklopedie. *Čárové kódy* [online]. Datum poslední revize 19.12.2012 [cit. 2013-01-05]. Dostupné na World Wide Web: <[http://cs.wikipedia.org/wiki/%C4%8C%C3%A1rov%C3%BD\\_k%C3%B3d](http://cs.wikipedia.org/wiki/%C4%8C%C3%A1rov%C3%BD_k%C3%B3d)>
- [12] KODYS. *Čárový kód* [online]. Praha: KODYS, spol. s.r.o., © 2009 [cit. 2013-01-10]. Dostupné na World Wide Web: <<http://www.kodys.cz/carovy-kod.html>>
- [13] DUBENG. *Čárové kódy* [online]. [cit. 2013-01-10]. Dostupné na World Wide Web: <<http://www.duben.org/skola/fel/5.rocnik/NM/TypyKodu2D.htm>>
- [14] KRČMÁŘ, Petr. *QR kódy* [online]. Datum publikování 1. 3. 2010 [online] [cit. 2013-01-10]. Dostupné na World Wide Web: <<http://www.root.cz/clanky/qr-kody-kilobajty-v-malem-obrazku/>>
- [15] WIKIPEDIE. *QR kód* [online]. Datum poslední editace 9. 1. 2013 [online] [cit. 2013-01-10]. Dostupné na World Wide Web: <[http://cs.wikipedia.org/wiki/QR\\_k%C3%B3d](http://cs.wikipedia.org/wiki/QR_k%C3%B3d)>
- [16] SOMMEROVÁ. *Základy RFID technologií: Výchovný materiál* [online]. Ostrava [cit. 2013-01-11]. Dostupné na World Wide Web: <[http://rfid.vsb.cz/miranda2/export/sites-root/rfid/cs/okruhy/informace/RFID\\_pro\\_Logistickou\\_akademii.pdf](http://rfid.vsb.cz/miranda2/export/sites-root/rfid/cs/okruhy/informace/RFID_pro_Logistickou_akademii.pdf)>. VŠB-TUO Hornicko-geologická fakulta.
- [17] DOLEŽAL, Luboš. *Dotkněte se RFID !: Úvod do technologie*. Praha, 2006 [online] [cit. 2013-01-11]. Dostupné na World Wide Web: <<http://www.rfid-epc.cz/dokumenty/>>



## **Seznam příloh**

- I. Třídní diagram
- II. QR kódy
- III. Bakalářská práce CD

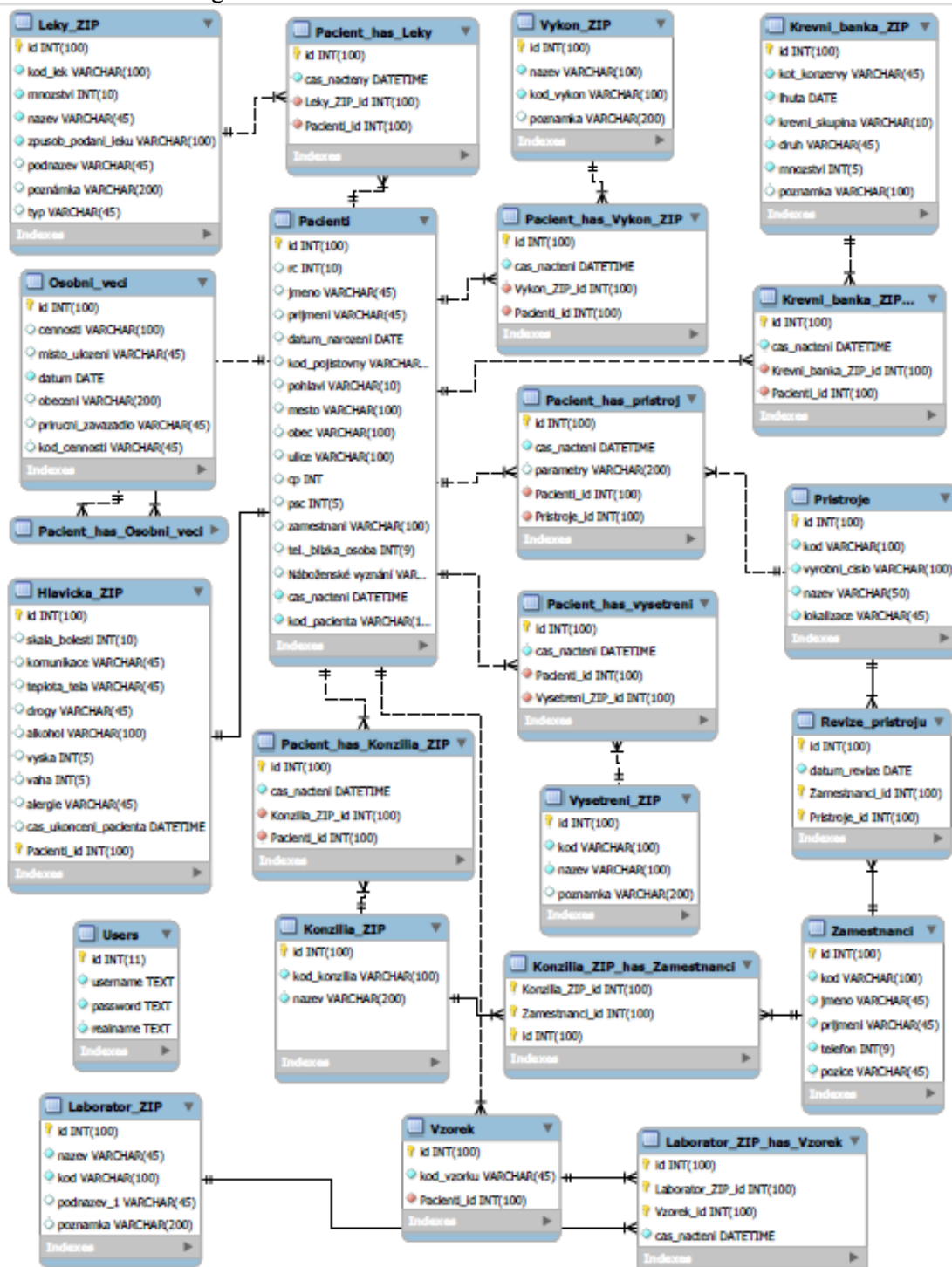
## **Obsah CD**

- I. Dokumenty – Bakalářská práce (gre0040\_BP.doc, gre0040\_BP.pdf)
  - Evidence přístrojů (Evidence přístrojů.pdf)
  - Třídní diagram (Třídní diagram.pdf)
  - QR kódy (QR kody.pdf)
- II. Aplikace – soubor InformaceNFC.zip

# Příloha

## I. Třídní diagram

## I. Třídní diagram



II. QR kódy

**QR kódy**



**Kod: 735ad635**

Pacient: id-214

jméno- Klárka

příjmení-Foltynova



**Kod: VYS0001**

Nazev: EKG